Can We Prevent Collusion in Multiplayer Online Games?

Jouni Smed Timo Knuutila Harri Hakonen

Department of Information Technology
FI-20014 University of Turku, Finland
{jouni.smed, timo.knuutila, harri.hakonen}@utu.fi

Abstract

Collusion is covert co-operation between participants of a game. It poses a serious problem to multiplayer games that do not allow the players to share knowledge or resources with other players. Fair play requires that players act independently to gain best possible individual result and are also able to analyse the action-reaction loop of the other players. In this paper, we discern two basic types of collusion – among the participants and among the players – and give a classification of different collusion types. Since collusion cannot be tackled using real-time methods, we propose how to counteract collusion using prevention and detection methods.

1 Introduction

Cheating in a computer game can be divided to technical exploitations (e.g. tampering with the network traffic or cracking the software) and rule violations (Smed and Hakonen, 2006, §10). Many games assume that the players are rivals and, therefore, the rules forbid collusion, where two or more opposing players co-operate covertly towards a common goal. Especially imperfect information games such as poker, where each player has access only to a limited amount of information, forbid collusion by sharing information. If the players are physically present, it is easier to detect any attempts of collusion (e.g. coughs, hand signals, or coded language), whereas the anonymity of multiplayer online games makes collusion detection a difficult problem.

The colluders can share knowledge or resources among themselves, which means that collusion can take many forms: Co-operating players can engage in soft play and refrain from attacking one another. A gang of players can ambush and rob other players in a role-playing game. A novice chess player can resort to an expert – human or computer program – to make better moves. Participants in a tournament can pre-arrange the outcome of their matches to eliminate other players. Players belonging to the same clan can send numerous and seemingly independent complaints to the administrator to get an innocent player banned. A friend participating as a spectator in a first-person shooter game can scout the arena and reveal the location of the enemies.

In this paper, we aim at recognizing different types of collusion. This enables us to discern what kind of methods could be used in preventing them. After reviewing the background, we go through examples of co-operation in *Age of Empires* and collusion in poker. This leads us to classify collusion types into three distinct categories and discuss how to counteract them. These countermeasures either try to prevent the collusion beforehand or detect it afterwards.

2 Background

The fundamental problem of collusion recognition is to discern *deliberate* collusion from mistakes, luck, and skill (or the lack of it). Therefore, collusion recognition has to observe the following factors:

- Learning curve: As a player learns to play the game, his tactics and strategies tend to get better.
- Familiar company: A player has often a better success when the game-partners are familiar to him.
- Sensible play: Each player tries to minimize the worst possible outcome of the game.
- Conflicting interest: Decisions and actions concerning multiple players contain often interests that can be, at the same time, mutually beneficial and conflicting depending on the perspective.

An advanced player usually considers and seeks for actions that provide predictability from his perspective. This means that he acts so that the number of possible outcomes of the whole game narrows down. To

achieve predictability he can aim at a stable situation in the game, and, to that end, players commonly develop repeating manoeuvres and routines that occur in certain situations (e.g. rules of thumb providing good a tactic or strategy). Another way to achieve predictability is to collude so that a (local) stability exists among the colluders.

A sensible player normally avoids wasting his time in unfruitful or unimportant conflicts and prefers agreements to preserve resources for some other purpose. Prolonged conflict has intentional (or non-operational) significance, because the effort to win has a cost (e.g. time, attention or resources required).

If two equal rivals engage in a conflict, they may know the equality beforehand, which leads to a cautious play. On the other hand, if equality is known only after the conflict, rivalry can be followed by tacit collusion (e.g. truce) until the status quo changes. If two rivals are not equal (in strength, resource types or knowledge) and do not engage in a conflict, we can suspect that they have formed a covert coalition.

2.1 Detecting Collusion

We conjecture that the smaller the set of choices the players face, the harder it is to hide the evidence of collusion. In card games, for example, the set of possible actions in a turn is limited, which means that it is difficult to evade detectable behavioural patterns. Since collusion is, fundamentally, one behavioural pattern, it is possible to develop methods for detecting it afterwards. For example, Rabiner (1989) describes a simple method for recognizing the tossing of a fake coin using hidden Markov models. Ercole et al. (2002) discuss how to capture colluders in a multiple choice examination. Also, the very fact that online poker game sites exist and thrive gives a real-world support to that it is possible to root out collusion in certain types of games – or at least give a believable threat of getting caught.

As the degree of freedom (i.e. the range of actions to choose from) increases, collusion detection becomes a tedious task. Especially online role playing game sites acknowledge the problem but admit that automatic policing is difficult to realize reliably. For example, *World of Warcraft* does not allow two rivaling guilds to collude and tries to detect it by tracking unusual patterns (Blizzard Entertainment, 2006).

Reliable collusion detection requires the following (True Poker, 2006):

- We can detect opportunities in order to recognize the possibilities for gain. The opportunities can be allowed by the rules or they can result from an unjust action.
- We can form a player profile by using pattern recognition. The profile concretizes different behaviours such as risk-taking, impatience, and favourite actions.
- We can analyse statistically the actions to determine the player's reputation. The reputation scheme can be open so that the players can rank one another. For example, a similar system is in use in Amazon.com where users can rank the products and the reviews of products, in eBay where the sellers are ranked, and in Xbox Live where TrueSkill system matches the players (Perry, 2005).
- We can analyse the play history to have a ground for penalization. By profiling the game players behaviour over time we can observe noticeable changes from usual habits which can indicate collusion.

All this analysis requires that we have some reference points such as:

- Game AI modelling expert play: How to value the actions and what is the best outcome that can be achieved with fair play?
- Game AI playing randomly: What is the least purposeful result?
- Player categorization: What is the typical selection of actions for a player?
- Player comparison: Are there correlations between the players' actions?

The reference point helps us in seeking out evidence of suspicious actions related to individual players or groups of players over time. We must also evaluate how much the action contributes to operational, tactical, and strategic level decision-making (i.e. the meaning and intention of a decision), and how rational the decision is in these levels (i.e. its degree of utility).

2.2 Classifications

Collusion classification is based on the type of agreement that the colluders have. The first categorization observes the *level of agreement*:

(i) Express collusion: The colluders have an explicit hidden agreement due to covert communication.

(ii) Tacit collusion: The colluders have no agreement nor covert communication but act towards a mutually beneficial goal such as attacking against the best player (to prevent him from

winning) or the worst player (to force him out of the game).

(iii) Semicollusion: Collusion is limited to certain decision types and otherwise the colluders compete against one another normally.

The second categorization is based on the *content of the agreement*, which can be:

(i) Concealed stance: The colluder adapts different play methods against co-colluders and other players. For example, the colluders can engage in soft play and refrain from offensive action against one another, or even lose deliberately in a tournament to enhance co-colluders' position.

(ii) Knowledge sharing: The colluder gets support for decision-making from an expert co-colluder, who is not necessarily taking part in the game as a player. For example, a veteran player can help a novice by hinting which weapon to use against a given enemy.

(iii) Information sharing: The colluder exchanges or trades information related to the current game situation. For example, the colluder can signal to the co-colluders the movements of enemy troops.

(iv) Resource sharing: The colluder receives, donates or trades game-related resources with the co-colluders. For example, the colluders can whip-saw to raise the costs in the in-game market to force out non-colluding players.

Collusion can also combine these aspects. For instance, players can agree to collude (i.e. express collusion) using both soft play (i.e. concealed stance) and signalling (i.e. information sharing).

3 Examples

Since collusion is covert co-operation, any form of co-operation is possible to use in collusion. In this section, we take first a look at a commercial computer game that allows the players to engage in co-operation in many different ways and levels. After that we review the work done on counteracting collusion in poker.

3.1 Co-operation in *Age of Empires*

To demonstrate the possible ways of co-operation let us use the widely-known real-time strategy game *Age of Empires II: The Age of Kings* (Ensemble Studios, 1999) as an example. The game comprises at maximum eight players, which battle against one another or form alliances. Since co-operation is not forbidden by the game rules, there is no collusion. Nevertheless, since the game provides the players with a rich set of tools for co-operation, it gives good examples of the types of co-operation (and collusion) that the players can engage in.

- Forming alliances: Players can form alliances which means that they cannot attack one another's troops. Alliance indicates that the players are fighting for a common goal, and it is a prerequisite for other forms of co-operation (e.g. sharing knowledge).
- Sharing knowledge: A player can research a technology called 'Cartography' which allows him to see everything that his alliances see in the game world.
- Donating resources: A player can donate a part of his resources to an allied player. For example, resource dumping can help the other player to develop quickly militarily, while the other focuses on gathering resources. Donation can be also done by repairing allied player's building or healing his troops.
- Sharing control: Two or more humans can control one player in the game.
- Providing intelligence: When a player is defeated or quits, the fog-of-war is removed and he can observe the whole game world in a spectator mode. Although he cannot affect the game world any more, he can communicate to the other players what he sees.

It should be noted that none of this is forbidden but the game actually encourages the players to co-operate.

3.2 Collusion in Poker

Mental poker, introduced by Shamir et al. (1981), is ordinary poker where the players play by exchanging messages (e.g. over the phone) instead of cards. Originally, the problem was ensuring a fair deal. Crépeau (1986, 1987) recognised first the problem of collusion in mental poker but did not provide any methods for solving it. Similarly, the later studies have focused on ensuring fair shuffling, dealing and betting and the problems caused by collusion have been brushed aside. For a review on mental poker protocols, see Castellà-Roca (2005).

Hall and Schneier (1997) study the dealing and betting in online casinos but omit collusion. Johansson

et al. (2003) use poker as an example to demonstrate that there are no pre-emptive nor real-time countermeasures against collusion. The organizer of an online game can try to track down the players, but that alone is not enough because the player's physical identity (i.e. his game account) does not reflect who is actually playing the game. Another approach is to analyse the game data to find out if there are players who participate often in the same games and, over a long period, profit more than they should (Vallvè-Guionnet, 2005). However, this kind of profiling requires a sufficient amount of game data, and collusion can be detected only afterwards.

Poker players can collude in two ways: In active collusion, colluding players play more aggressively than they normally would (e.g. outbet non-colluding players). In passive collusion, colluding players play more cautiously than they normally would (e.g. only the one with the strongest hand continues while the others fold). Active collusion can be detected afterward by analysing the game data, but it is next to impossible to discern passive collusion from a cautious normal play (Johansson et al., 2003).

Online poker site terms and rules usually stipulate that anyone attempting to collude will be prohibit permanently using the services provided by the site and their account will be terminated immediately; for example, see Noble Poker (2006, §6.8) and Titan Poker (2006, §6.8). Collusion detection is mainly based on investigating complaints from other players, although some sites use methods for analysing the game data to find play patterns typical of collusion (True Poker, 2006).

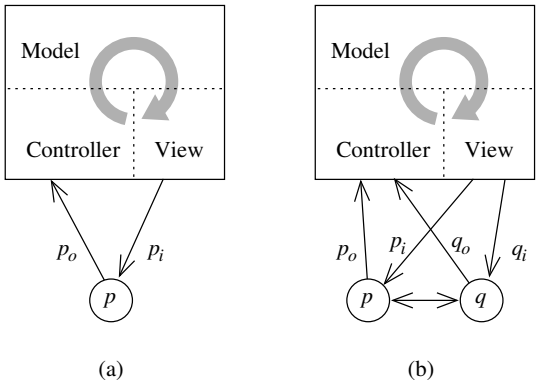

Figure 1: Model–View–Controller and collusion. (a) Participant p has input p_i and output p_o, which him a player in the game. (b) When participants collude, they both can affect the game situation individually.

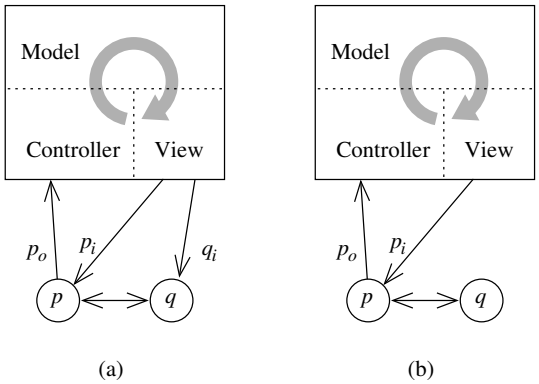

Figure 2: (a) A participant has only a view to the game and can affect the game situation indirectly through a player. (b) A participant can communicate directly to a player.

4 Roles in Collusion

To discern different types of collusion, we must observe that the relationship between *players* and *participants* of the game is not one-to-one. For example, if two human participants take turns in controlling one game character, they appear in the game as one player. Similarly, a human can take part in the game in a spectator mode, where he is not a player but can participate by observing other players. In contrast, the player can also be a synthetic player, which is a computer-controlled entity that does not correspond to any human participant. Consequently, we must differentiate

(i) collusion among the players of the game, and
(ii) collusion among the participants of the game.

To clarify the difference between these two types of collusion, let us look at the situation using Model–View–Controller (MVC) architectural pattern. The basic idea of MVC is that the representation of the underlying application properties (Model) should be separated from the way they are presented to the user (View) and from the way the user interacts with them (Controller); for a detailed discussion, see Smed and Hakonen (2006, §1.1). Figure 1(b) illustrates two colluders, p and q. If they both take part in the game as players, they both can affect the game individually (i.e. they both have access to the View and Controller part). In Figure 2, q is a non-playing participant, who has access only to the View part either directly (e.g. as a spectator) or indirectly (e.g. through p).

When players collude, they share and utilize knowledge or resources inside the game. This means that it is possible to detect the collusion, since the organizer of the game can analyse if the behaviour of play-

ers diverges from what can be reasonably expected. Conversely, detecting colluding participants is much harder, because it requires analysing the behaviour of individual players.

In the following subsections, we divide collusion types based on how the collusion works instead of what for it works. We describe collusion types, their indicators, and possible countermeasures.

4.1 Participant Identity Collusion

Let us first take a look at how a single player is perceived to participate in a game; see Figure 1(a). One of the simplest way to collude is to change or boost the participant that controls the player. Normally, we require that there is a unique human in the player loop. This concerns the relationship between a single player and the participant, and, thus, this collusion can be present in all other forms of collusion. Depending on how many parties are concerned, this kind of covert play can take the forms of player controller collusion or self-collusion.

4.1.1 Player Controller Collusion

When a player is not controlled only by a single human participant, the collusion takes the form of shared control. In the simplest form this can be achieved by replacing p by some other human participant (e.g. an expert player). A sweatshop (i.e. continuous play by taking turns) manufacturing digital game items or game avatars is an example this kind of collusion. To counteract a sweatshop, player's behaviour in a game session or in certain situations can be monitored and analysed in real-time (Furnell and Dowland, 2000). This kind of intrusion monitor techniques provide a way to verify the physical identity using the player account. Naturally, a simple analysis of the session logs can reveal if any of the participants has an exceptional metabolism.

Alternatively, the decision-making can be delegated so that p becomes a computer-controlled entity, a synthetic player. Game design can be used to make it harder for synthetic players to take part in the game. For this purpose, the game world should not have static or repeating elements, which can be utilized, for example, by scripting. Also, CAPTCHA methods (Golle and Ducheneaut, 2005) could provide a preventive way to tell apart human and synthetic players. The idea behind these methods is to have 'public' steganography for synthetic players that are easy for humans to solve.

In complex form, player controller collusion includes several parties (see Figure 2). For instance, the participant p can get help from a human helper q, who augments p's capabilities to play. Alternatively, the helper can be an automatized tool such as an auxiliary information tool, a decision aid, or an external simulator and analyser. The helper can also be the front-end for the participant such as a reflex augmentator or an action enhancer. These tools give the player abilities that he would not normally have. Such automatized tools can be detected by observing monotonic and repetitive action chains that do not take into account the nuances of the game situation.

4.1.2 Self-collusion

If a single participant controls multiple players (i.e. has alias players), he is engaging in self-collusion. This situation is illustrated in Figure 1(b), when $p = q$. Since it is easier for one participant to fake multiple players than multiple participants to pretend to be one player, the intrusion monitoring methods do not necessarily work in this case. Instead of preventing, we can model the player's decision-making process and measure deviations from it. However, this detection approach must account for the effect of the learning curve.

If we assume that there is no learning (i.e. the decision-making process is static), it is possible to apply traditional pattern recognition methods to the game data. For example, game state relationships can be modelled with self-organizing maps (Kohonen, 1995) and time series can be adapted using hidden Markov models (Rabiner, 1989). When the participant learns more about the game his decision-making process changes. To take this into account, the modelling phase makes use of typical learning curves and participant profiles. To ease the task further, the player types can be limited by offering dedicated servers for specific gaming types or target groups.

4.2 Inter-player Collusion

The second collusion type concerns multiple different participants as illustrated in Figure 1(b), when $p \neq q$. We can differentiate three forms of this collusion type: spectator collusion, assistant collusion, and association collusion.

4.2.1 Spectator Collusion

A spectating player, who participates in the game as a ghost, can reveal extra knowledge to another player. In other words, we have the situation illustrated in

Figure 2(a), where p_i is not congruent to q_i in terms of game playing.

Since the spectator mode can be used to gather intelligence, we can easily prevent it by providing the spectators a delayed feed of the game. Suppose the delay is one minute, which means that the spectators see the game events that have already happened and not the current situation. This reduces the value of the observations of dynamic information (e.g. the position of enemies). Naturally, the static information such as the number of enemies or the geography of the game world, has some value for the active players. Technically this approach means that the game server has to send two feeds – interactive live feed for the players and delayed feed for the spectators – which yields more network traffic. However, the protocol of the delayed feed can optimized with message aggregation and compression (Smed and Hakonen, 2006, §9.2). Also, the delayed feed can serve as an input for other desirable game functionalities, including recording, monitoring and verifying distributed control, and realizing public collusion detection.

4.2.2 Assistant Collusion

A player can be advised by another player who does not aim at winning but assist as a (possibly altruistic) sidekick. This situation is illustrated in Figure 2(a), when p_i is congruent to q_i, and q_0 is non-existent in practice. For example, if the team players in a first-person shooter can communicate verbally, a scout with binoculars can direct how the assaults should advance towards the enemy but do not otherwise engage in fighting. More extreme and direct example of this type of collusion is to join the enemy and act as a spy.

This kind of collusion takes form in miraculous escapes from situations that are caused by a failed tactic. Also, this kind of collusion can be used to reveal and neutralize opponent's tactical advantages.

Because the colluders act as players in the game, illicit assistance is hard to prevent. However, because the input feeds of the players are known, it is possible to deduce which player has the information that is possibly utilized by another player. Also, to test the suspected players, the game can set up sting operations or game playing traps by generating customized situations that are first observed by them.

Note that if q gets cut off from the game, he can continue assisting if p's input feed is forwarded to him, as in Figure 2(b). This forwarding can be as simple as sharing the same monitor display. In this case, collusion changes to player controller collusion.

4.2.3 Association Collusion

Active players can co-operate secretly due to shared interests or efforts, see Figure 1(b), where p_i is congruent to q_i. The difference to assistant collusion is that colluding players are in a symbiotic relationship and collusion provides them with a way to achieve their individual goals. Because cheating does not respect protocols or other rules, this form of collusion is almost impossible to prevent without a physical supervision of participants. In theory, it is also hard to detect it afterwards. However, collusion always produces patterns between the states and actions, and the simpler the game is, the harder it is to avoid leaving or hiding these traces. In other words, there is a limit when avoiding getting caught becomes more tedious than the game itself. Thus, if the collusion detection is not allowed to become a game in itself (e.g. by penalization), the situation is not so pessimistic in practice.

The association collusion can exist when a player carries out pre-emptive actions without any possibility to observe the tactical situation. For example, he can load his gun just in time, he can select a suitable weapon arsenal before the situation actually changes, or he can suddenly change direction to engage the opponents. The collusion produces also simultaneous actions and correlating tactics. For example, the players can specialize so that they complement each others.

The main difficulty of association collusion is that there is no way to prevent the ingenious use of external communication channels. Nevertheless, it is possible to reduce the benefit of these external channels from various aspects.

In the *game content* aspect, the first impression can be renewed by naming and labelling the game elements differently for each game instance. This idea is present in *Nethack* (DevTeam, 2006), where the player must use effort to re-learn the effect of the game items.

In the *player profile* aspect, the intentions can be conceptualized by incorporating them into the game GUI: The player must declare his stance towards the other players and make his tactical or strategic decisions explicitly predefined before the actions. The player profile have two immediate benefits: Firstly, the game play can be made smoother by making the actions context sensitive. Secondly, it gives name for each intention, which reference concepts for post-analysis programs.

In the *timing* aspect, the response window can be so short that the player has no time to discuss and analyse the situation with other players. Also, it is

possible to bias the time-related phenomena within the limits of causality. For example, the game system can add jitter to communication to prevent use of timing covert channel (Yan, 2003).

In the *player identity* aspect, the player's identity can be obscured from the other players. The player can customize his avatar but the other players cannot see it modified. Also, the teams can be randomized, although this prevents co-operative play. This forces the colluding players to identity themselves to co-colludes inside the game world with an explicit signal.

In the *surveillance* aspect, if the colluding players are humans, the supervisors can also be humans. Actually, this introduces a higher level game between the colluders and the supervisors that some players may find more interesting. This calls for easy, public and open monitoring, where any participant can access the actions of some players. In this way the colluder do not know who supervises him and by what means and methods. The spectator feed is obvious mechanism to include this openness of decision-making and actions.

The transparency of actions must always be supplemented by substantial penalization (e.g. banning). To prevent collusion on penalization, there have to be a mechanism for the accused so that he can explain the suspicious actions later. It is worth noting that the required mechanisms already exist, since ordinary players commonly share experiences with friends and game community on online sites.

The surveillance can also include physical monitoring (web camera). This can be attached to reputation so that, for example, a player have to renew gained expertise level and reputation ranking by participating to supervised game instances from time to time (as in sports).

4.3 Game Instance Collusion

Collusion can be beneficial between the participants of different game instances, or between participants and administrators.

4.3.1 Multigame Collusion

The players of different game instances can also cooperate, as depicted in Figure 3(a). There are three typical intentions: Firstly, player q can provide intelligence to player p about the properties of the game world in general. For example, q can reveal secret game world locations or weaknesses of the synthetic players. Secondly, a group of players can scout in parallel different game servers to find a game instance

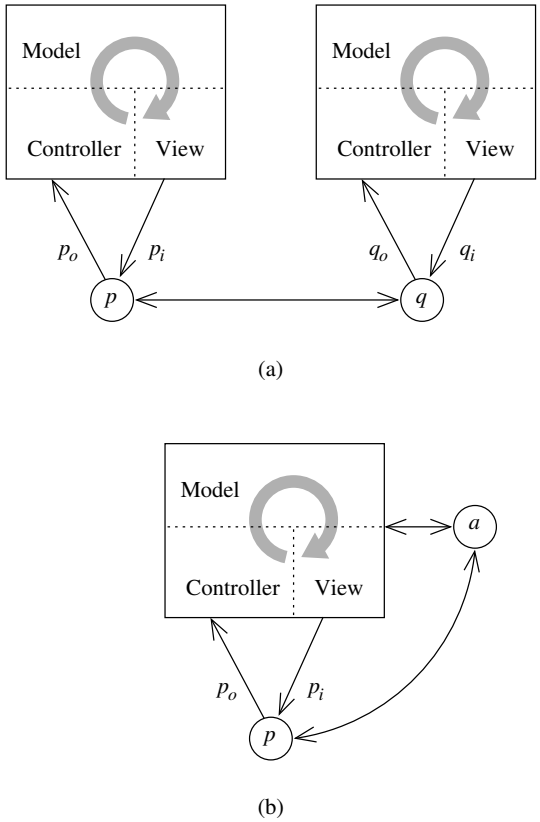

Figure 3: (a) Collusion can span over multiple game instances. (b) A player can collude with a game administrator.

that suits the group, and, after that, the whole group joins that game instance. Thirdly, especially when the game instances are not run in parallel, the players can collude on the tournament by fixing the match results together. Murdoch and Zieliński (2004) discuss more about collusion in tournaments.

4.3.2 Insider Collusion

A player can be advised by the game administrator or the game developer. The situation is illustrated in Figure 3(b). This collusion can be very unintentional and tacit. For example, when a player calls for product support, the help desk can reveal accidently something on the game world.

5 Conclusion

In the title, we asked whether collusion can be prevented in multiplayer online games. To break down this question, we can now answer that it is possible – either by making it difficult to engage in collusion

in the first place or by increasing the risk of getting caught afterwards. Consequently, the countermeasures are based on either prevention or detection. Their benefits are immediate: The players in multiplayer online games could be sure that there are no illicit alliances, gangs, clans or co-operation among rivals but everybody would have a fair playing field, which is one of the main goals of cheating prevention.

We also recognize that there is arising need for a third-party organization that grants and manages player licenses. Zetterström (2005) expresses a similar view and calls for an anti-cheating organization for online multiplayer games equivalent to the anti-doping organization WADA in sports. Such an organization could provide authentication of players and monitor their progress (e.g. maintain player rankings) but it would also provide a way to weed out players that systematically break the rules.

Acknowledgements
The authors thank Pyry Hakonen for pointing out collusion using parallel server scouting.

References

Blizzard Entertainment. Honor system Q&A with Kalgan. Web page, accessed April 28, 2006. ⟨http://forums.worldofwarcraft.com/thread.aspx?fn=blizzard-archive&t=24&p=1&tmp=1⟩.

J. Castellà-Roca. *Contributions to Mental Poker*. PhD thesis, Universitat Autònoma de Barcelona, Barcelona, Spain, 2005.

C. Crépeau. A secure poker protocol that minimizes the effect of player coalitions. In *Advances in Cryptology: Proceedings of Crypto '85*, volume 218 of *Lecture Notes in Computer Science*, pages 73–86. Springer-Verlag, 1986.

C. Crépeau. A zero-knowledge poker protocol that achieves confidentiality of the player's strategy or how to achieve an electronic poker face. In *Advances in Cryptology: Proceedings of Crypto '86*, volume 263 of *Lecture Notes in Computer Science*, pages 239–247. Springer-Verlag, 1987.

DevTeam. *NetHack 3.4.3*. 2006. ⟨http://www.nethack.org/⟩.

Ensemble Studios. *Age of Empires II: The Age of Kings*. Microsoft Games, 1999.

A. Ercole, K. D. Whittlestone, D. G. Melvin, and J. Rashbass. Collusion detection in multiple choice examinations. *Medical Education*, 36(2):166–172, 2002.

S. M. Furnell and P. S. Dowland. A conceptual architecture for real-time intrusion monitoring. *Information Management & Computer Security*, 8(2): 65–74, 2000.

P. Golle and N. Ducheneaut. Preventing bots from playing online games. *Computers in Entertainment*, 3(3):3–3, 2005. ⟨http://doi.acm.org/10.1145/1077246.1077255⟩.

C. Hall and B. Schneier. Remote electronic gambling. In *13th Annual Computer Security Applications Conference*, pages 227–230, San Diego, CA, USA, December 1997.

U. Johansson, C. Sönström, and R. König. Cheating by sharing information—the doom of online poker? In *Proceedings of the 2nd International Conference on Application and Development of Computer Games*, pages 16–22, Hong Kong SAR, China, January 2003.

T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, Berlin, Germany, 1995.

S. J. Murdoch and P. Zieliński. Covert channels for collusion in online computer games. In *Information Hiding: 6th International Workshop*, volume 3200 of *Lecture Notes in Computer Science*, pages 355–369, Toronto, Canada, May 2004. Springer-Verlag.

Noble Poker. Terms and conditions. Web page, accessed April 27, 2006. ⟨http://www.noblepoker.com/termsofuse.html⟩.

D. C. Perry. Live in the next generation: The TrueSkill system, 2005. ⟨http://xbox360.ign.com/articles/662/662347p1.html⟩.

L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

A. Shamir, R. L. Rivest, and L. M. Adleman. Mental poker. In *The Mathematical Gardner*, pages 37–43. Prindle, Weber & Schmidt, Boston, MA, USA, 1981.

J. Smed and H. Hakonen. *Algorithms and Networking for Computer Games*. John Wiley & Sons, Chichester, UK, 2006.

Titan Poker. Terms and conditions. Web page, accessed April 27, 2006. ⟨http://www.titanpoker.com/termsofuse.html⟩.

True Poker. Anti-collusion, fraud detection, and random card shuffling. Web page, accessed April 26, 2006. ⟨http://www.truepoker.com/anti_collusion.html⟩.

C. Vallvè-Guionnet. Finding colluders in card games. In *Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05)*, volume II, pages 774–775, Las Vegas, NV, USA, April 2005.

J. Yan. Security design in online games. In *Proceedings of the 19th Annual Computer Security Applications Conference (ACSAC'03)*, pages 286–297, Las Vegas, NV, USA, December 2003.

J. Zetterström. A legal analysis of cheating in online multiplayer games. Master's thesis, School of Economics and Commercial Law, Gothenburg, Sweden, 2005.